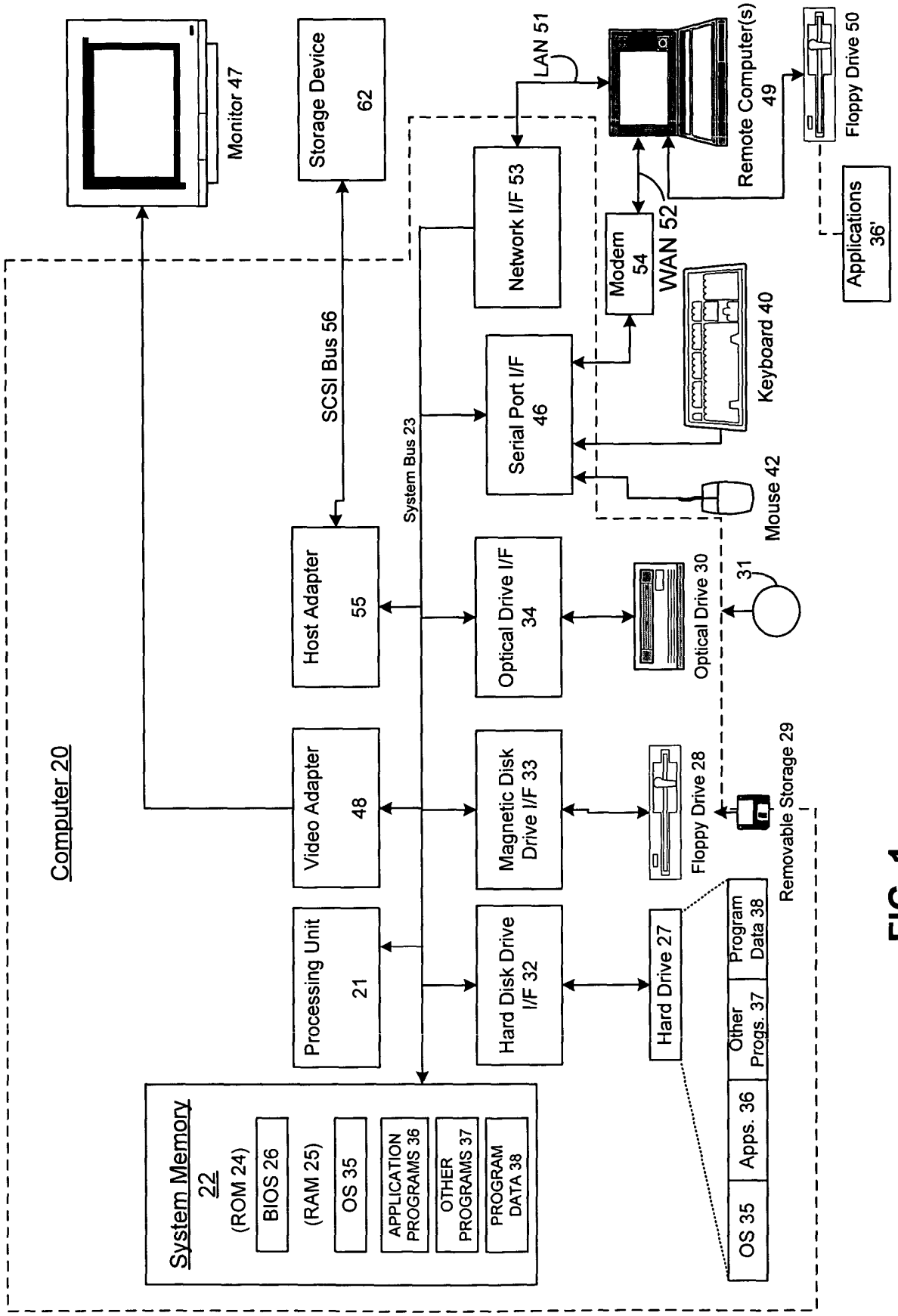

Computer 20
Monitor 47
Storage Device 62
LAN 51
Remote Computer(s) 49
Floppy Drive 50
Applications 36'
Network I/F 53
Modem 54
WAN 52
SCSI Bus 56
System Bus 23
Serial Port I/F 46
Keyboard 40
Mouse 42
Host Adapter 55
Optical Drive I/F 34
Optical Drive 30
31
Video Adapter 48
Magnetic Disk Drive I/F 33
Floppy Drive 28
Removable Storage 29
Processing Unit 21
Hard Disk Drive I/F 32
Hard Drive 27
Other Progs. 37
Program Data 38
OS 35
Apps. 36
System Memory 22
(ROM 24)
BIOS 26
(RAM 25)
OS 35
APPLICATION PROGRAMS 36
OTHER PROGRAMS 37
PROGRAM DATA 38

FIG. 1

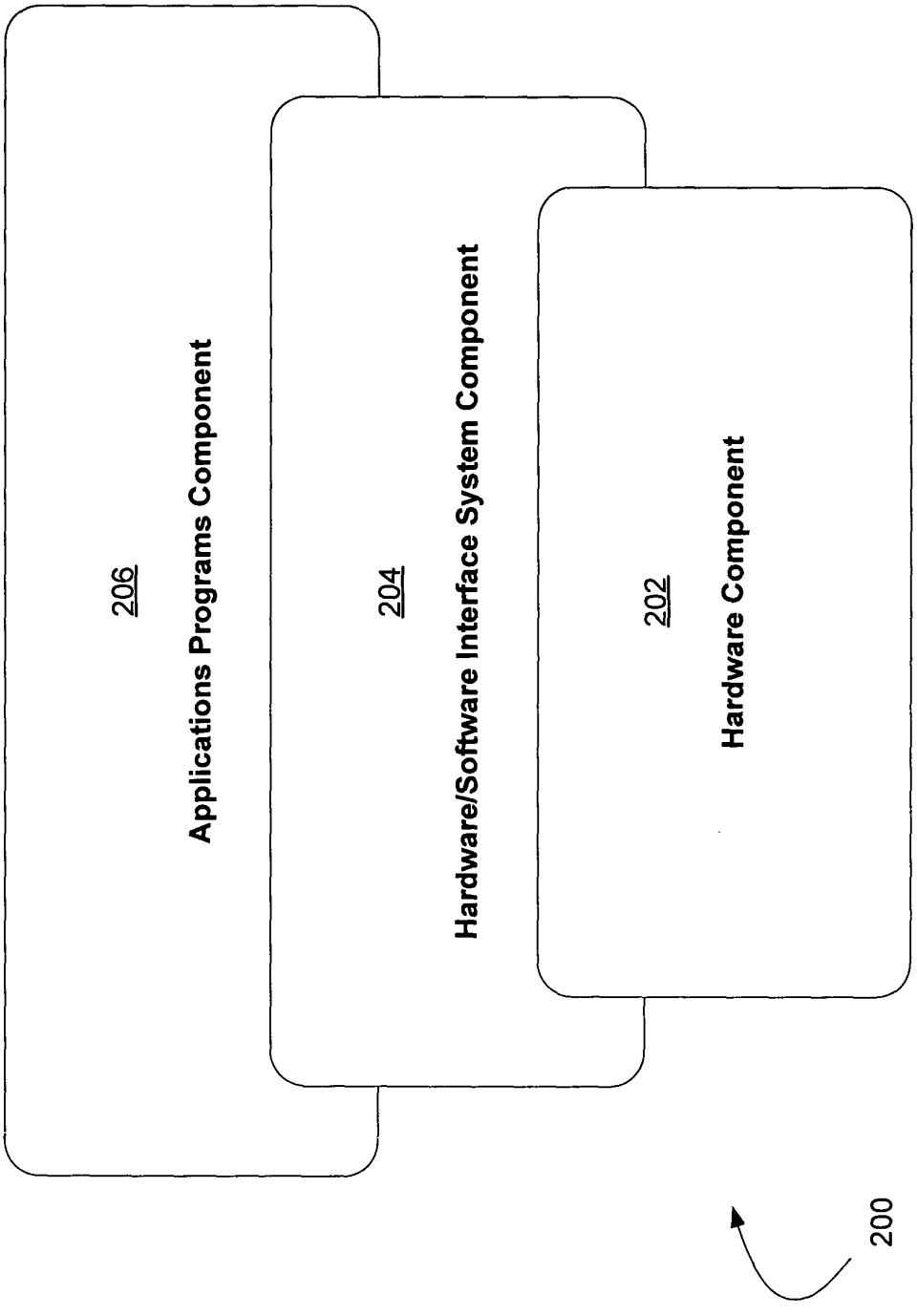
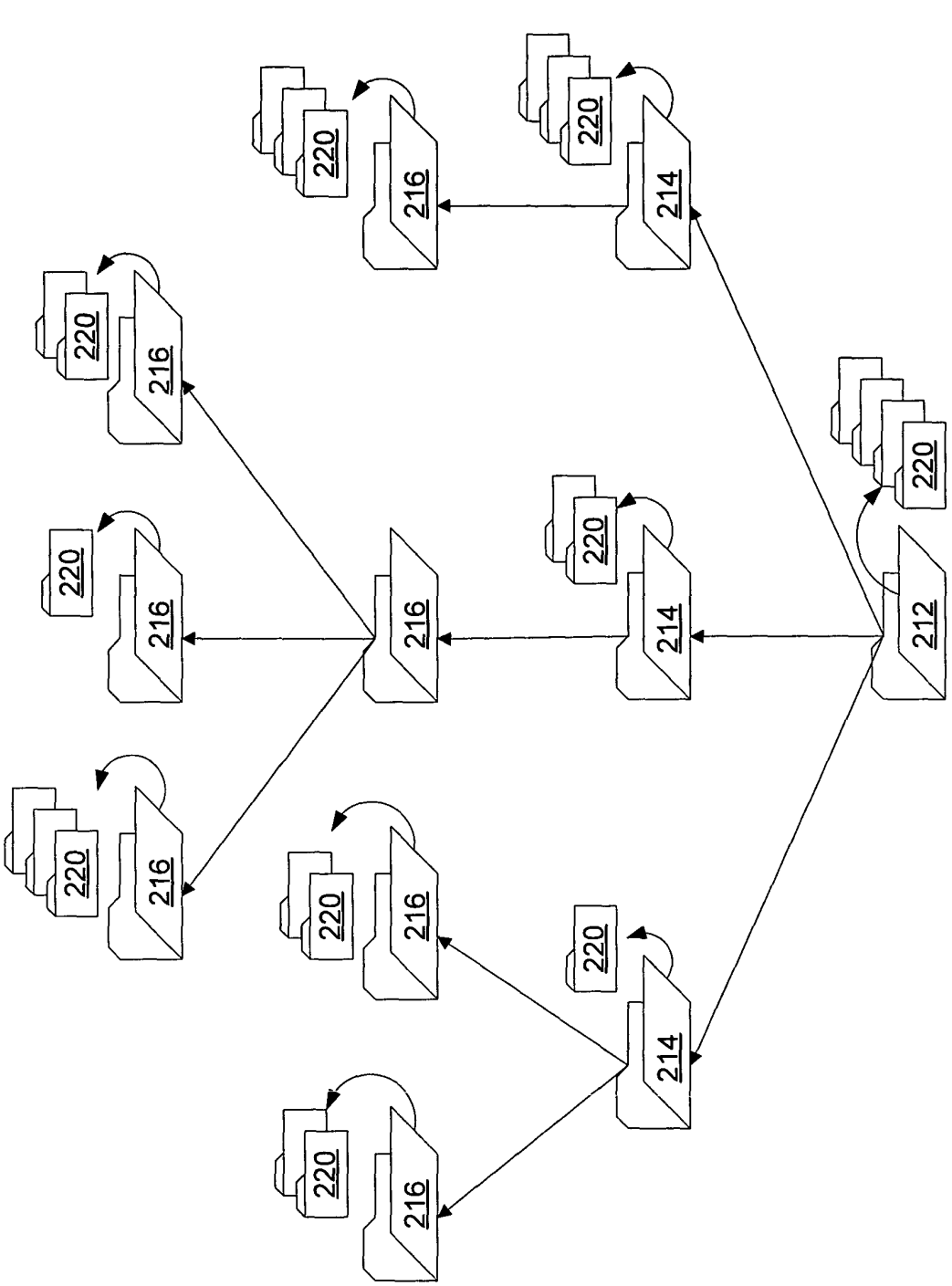

206
Applications Programs Component
204
Hardware/Software Interface System Component
202
Hardware Component
200

FIG. 2

FIG. 2A



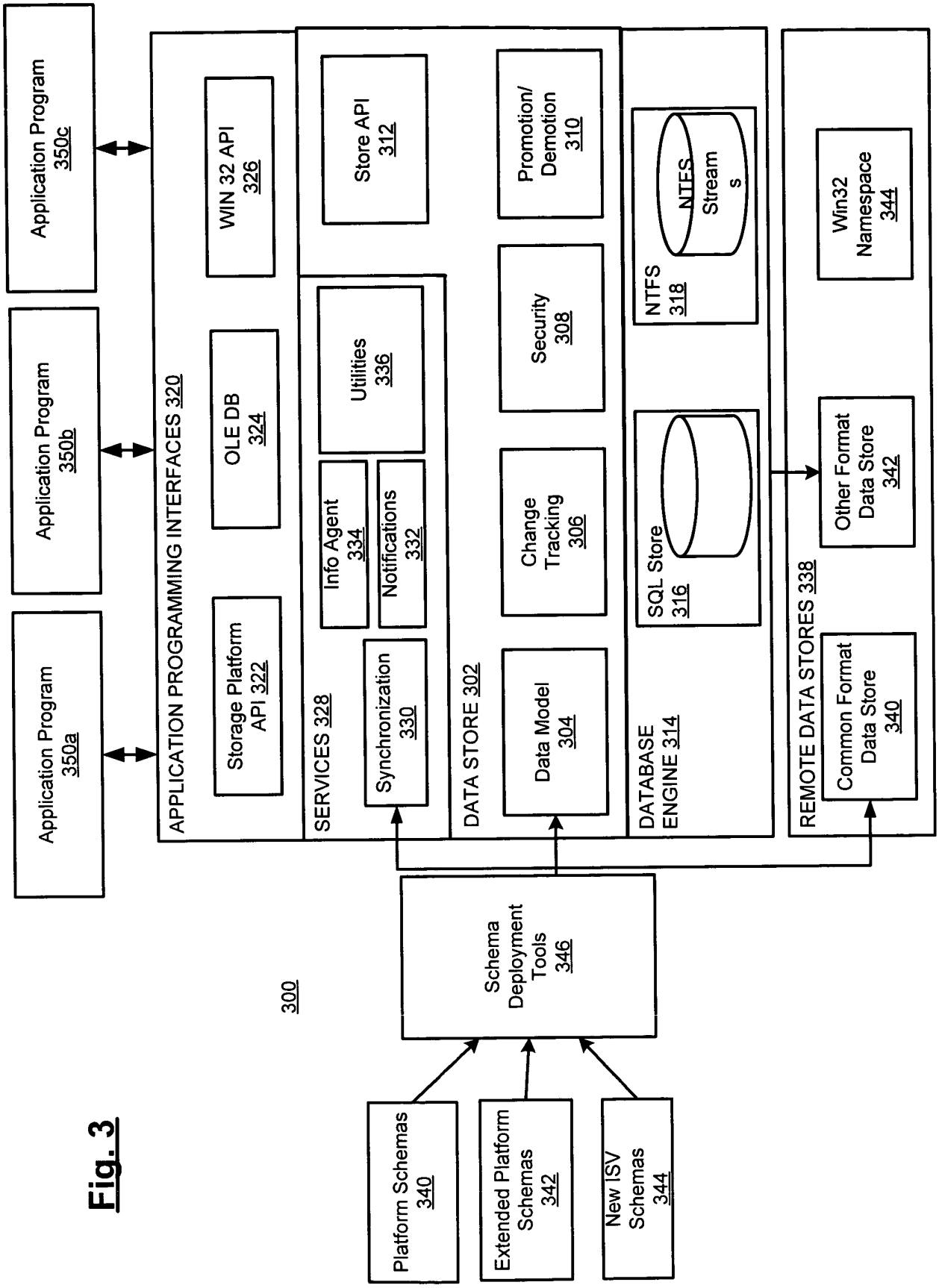

Application Program 350c
Application Program 350b
Application Program 350a
APPLICATION PROGRAMMING INTERFACES 320
WIN 32 API 326
OLE DB 324
Storage Platform API 322
SERVICES 328
Store API 312
Utilities 336
Info Agent 334
Notifications 332
Synchronization 330
DATA STORE 302
Promotion/ Demotion 310
Security 308
Change Tracking 306
Data Model 304
DATABASE ENGINE 314
NTFS 318
NTFS Stream s
SQL Store 316
REMOTE DATA STORES 338
Win32 Namespace 344
Other Format Data Store 342
Common Format Data Store 340
300
Schema Deployment Tools 346
Platform Schemas 340
Extended Platform Schemas 342
New ISV Schemas 344

Fig. 3

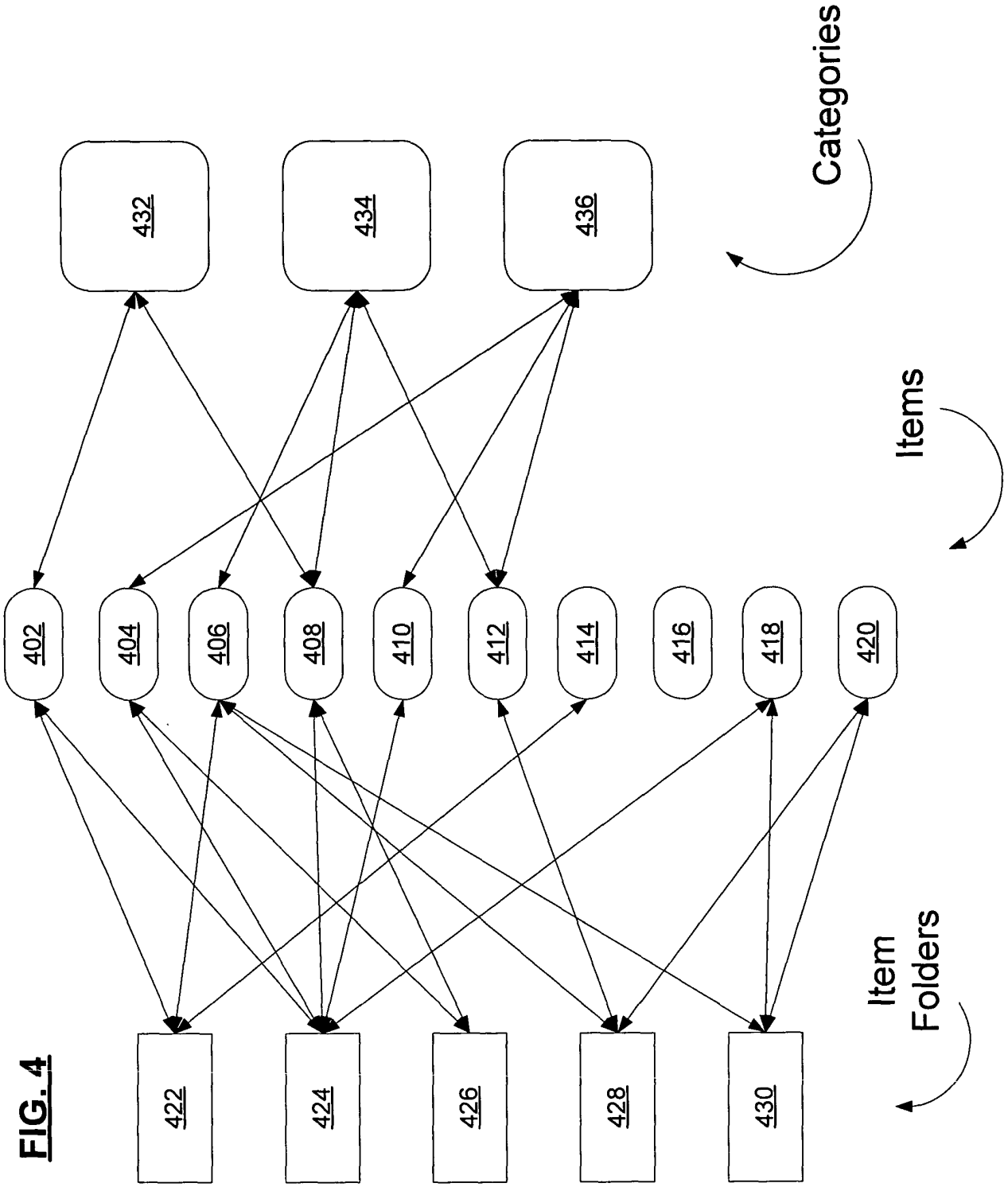

Categories
Items
Item Folders
432
434
436
402
404
406
408
410
412
414
416
418
420
422
424
426
428
430

FIG. 4

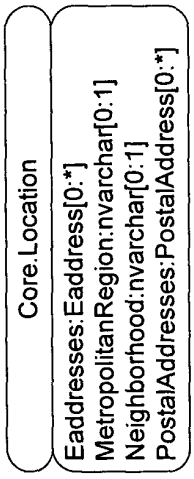
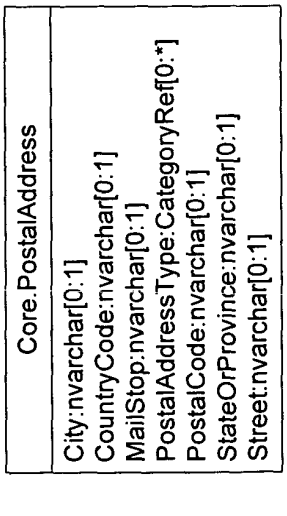


FIG. 5A



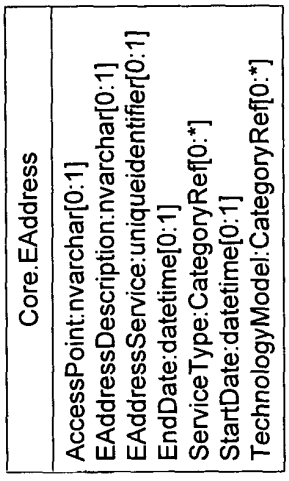


FIG. 5B

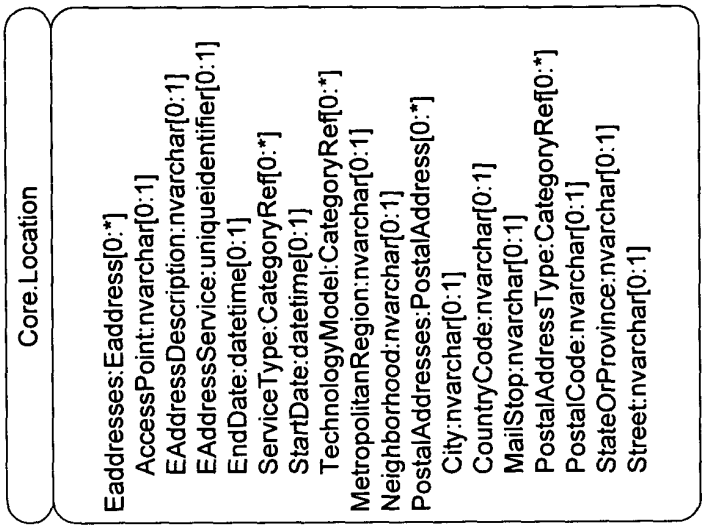


FIG. 5C

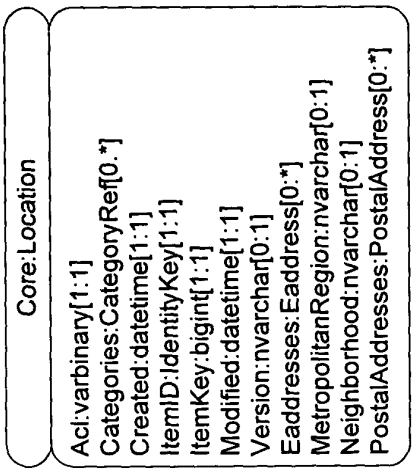
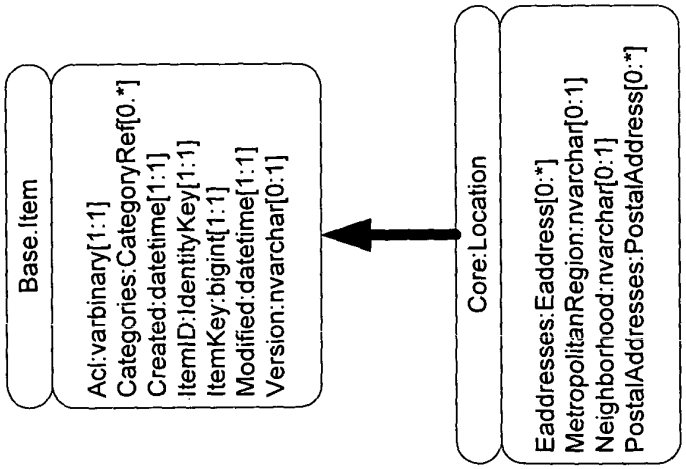

Core:Location
Acl:varbinary[1:1]
Categories:CategoryRef[0:*]
Created:datetime[1:1]
ItemID:IdentityKey[1:1]
ItemKey:bigint[1:1]
Modified:datetime[1:1]
Version:nvarchar[0:1]
Eaddresses:Eaddress[0:*]
MetropolitanRegion:nvarchar[0:1]
Neighborhood:nvarchar[0:1]
PostalAddresses:PostalAddress[0:*]

FIG. 6B

FIG. 6A


Base.Item
Acl:varbinary[1:1]
Categories:CategoryRef[0:*]
Created:datetime[1:1]
ItemID:IdentityKey[1:1]
ItemKey:bigint[1:1]
Modified:datetime[1:1]
Version:nvarchar[0:1]
Core:Location
Eaddresses:Eaddress[0:*]
MetropolitanRegion:nvarchar[0:1]
Neighborhood:nvarchar[0:1]
PostalAddresses:PostalAddress[0:*]

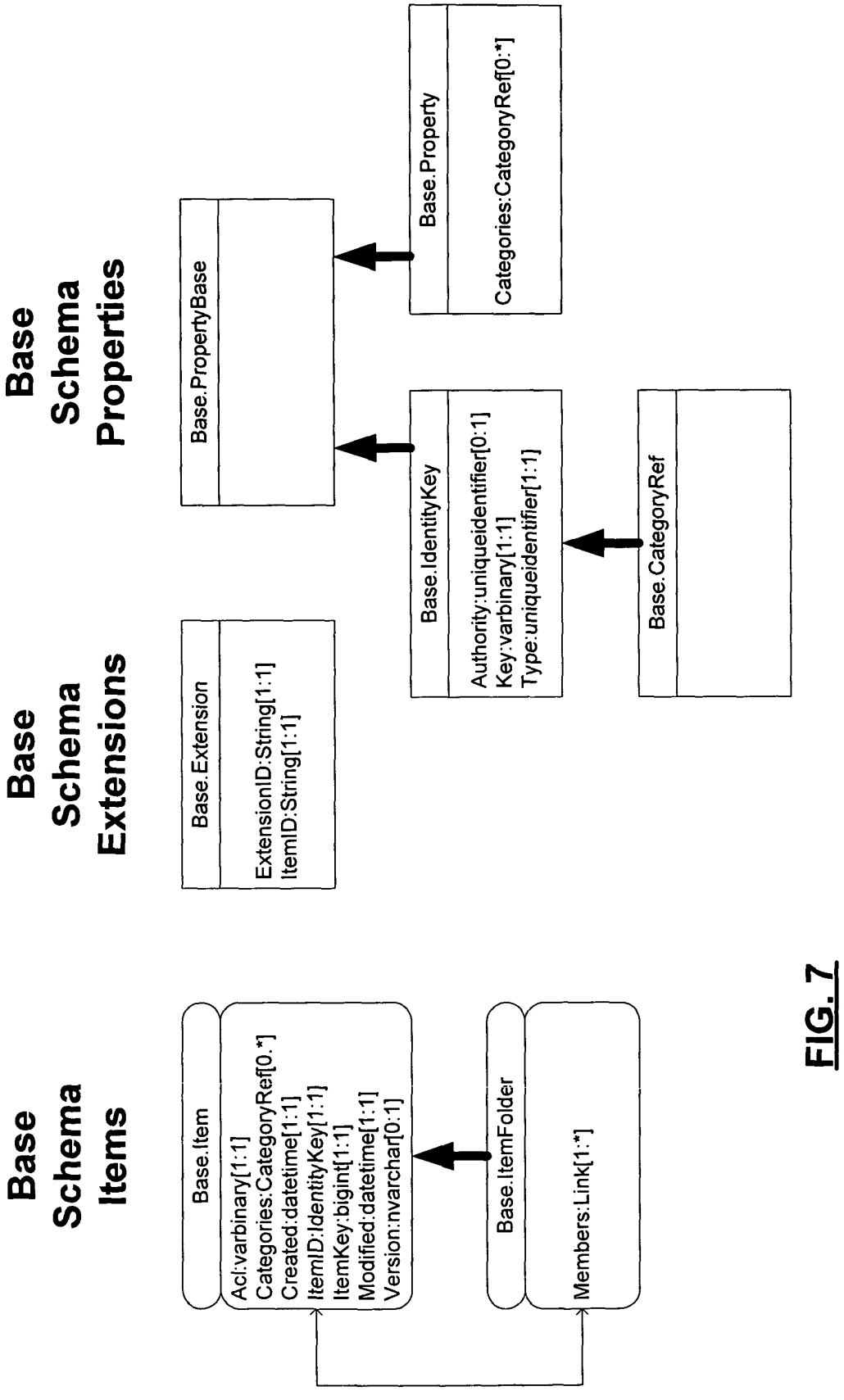

Base Schema Items
Base.Item
Acl:varbinary[1:1]
Categories:CategoryRef[0:*]
Created:datetime[1:1]
ItemID:IdentityKey[1:1]
ItemKey:bigint[1:1]
Modified:datetime[1:1]
Version:nvarchar[0:1]
Base.ItemFolder
Members:Link[1:*]
Base Schema Extensions
Base.Extension
ExtensionID:String[1:1]
ItemID:String[1:1]
Base.IdentityKey
Authority:uniqueidentifier[0:1]
Key:varbinary[1:1]
Type:uniqueidentifier[1:1]
Base.CategoryRef
Base Schema Properties
Base.PropertyBase
Base.Property
Categories:CategoryRef[0:*]

FIG. 7

Docket No.: MSFT-1748/302722.01 App. No.: Not Yet Assigned Filed: Herewith
Title: SYSTEMS AND METHODS FOR REPRESENTING UNITS OF INFORMATION MANAGEABLE BY A HARDWARE/SOFTWARE INTERFACE SYSTEM BUT INDEPENDENT OF PHYSICAL REPRESENTATION
Inventors: J. Patrick Thompson et al.
Attorney: Richard W. Knight Phone: (215) 568-3100

FIG. 8A

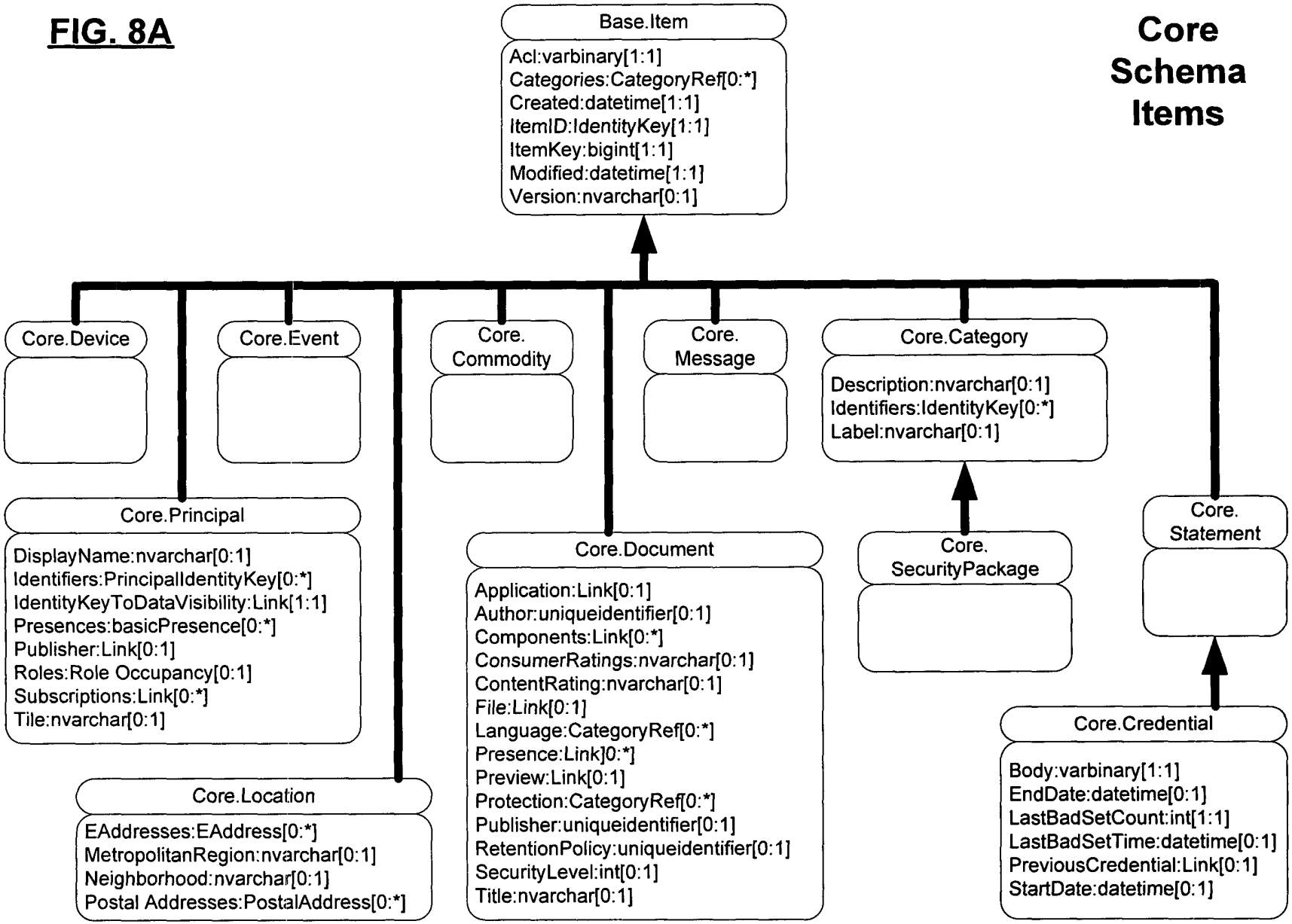
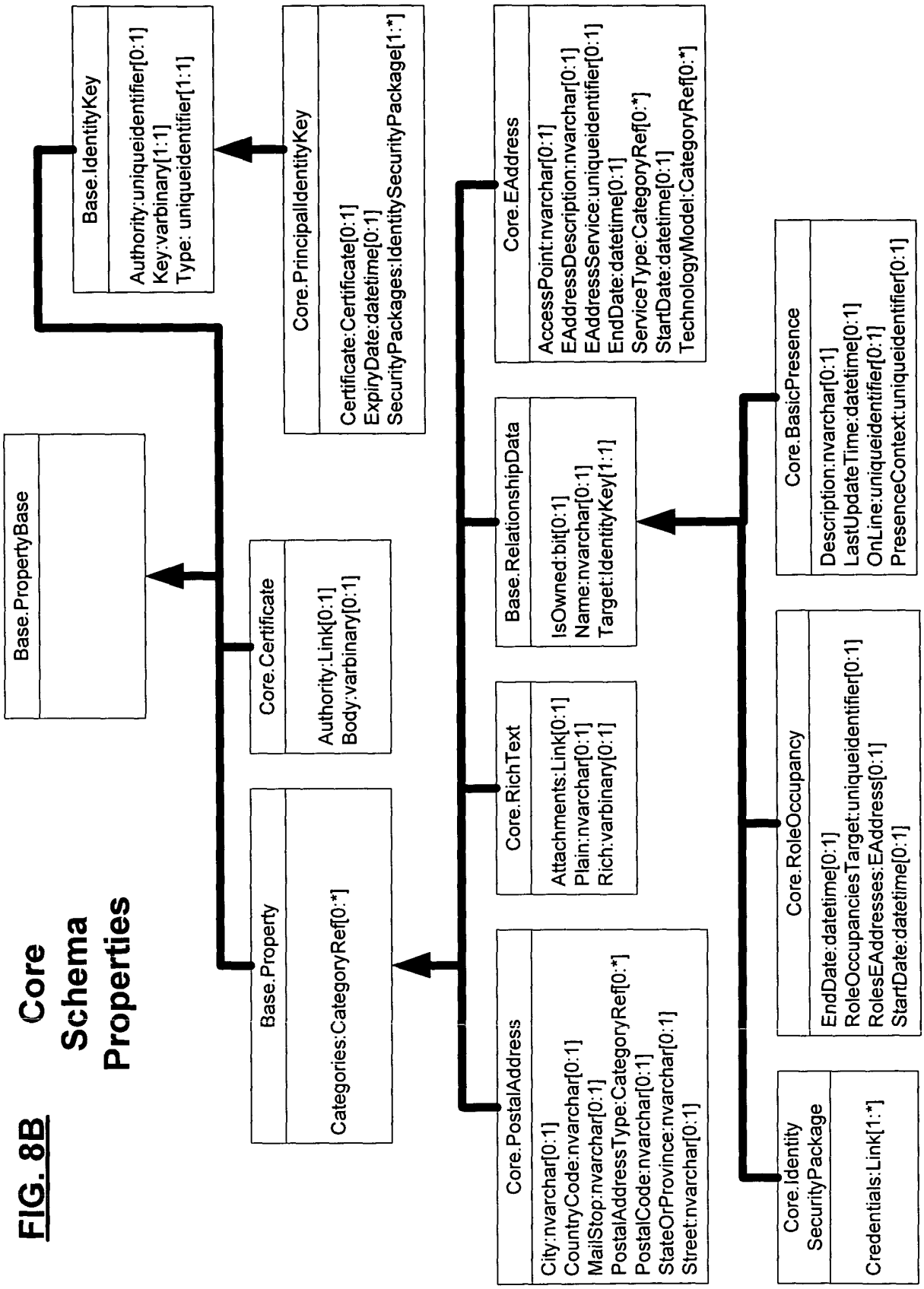


FIG. 8B Core Schema Properties



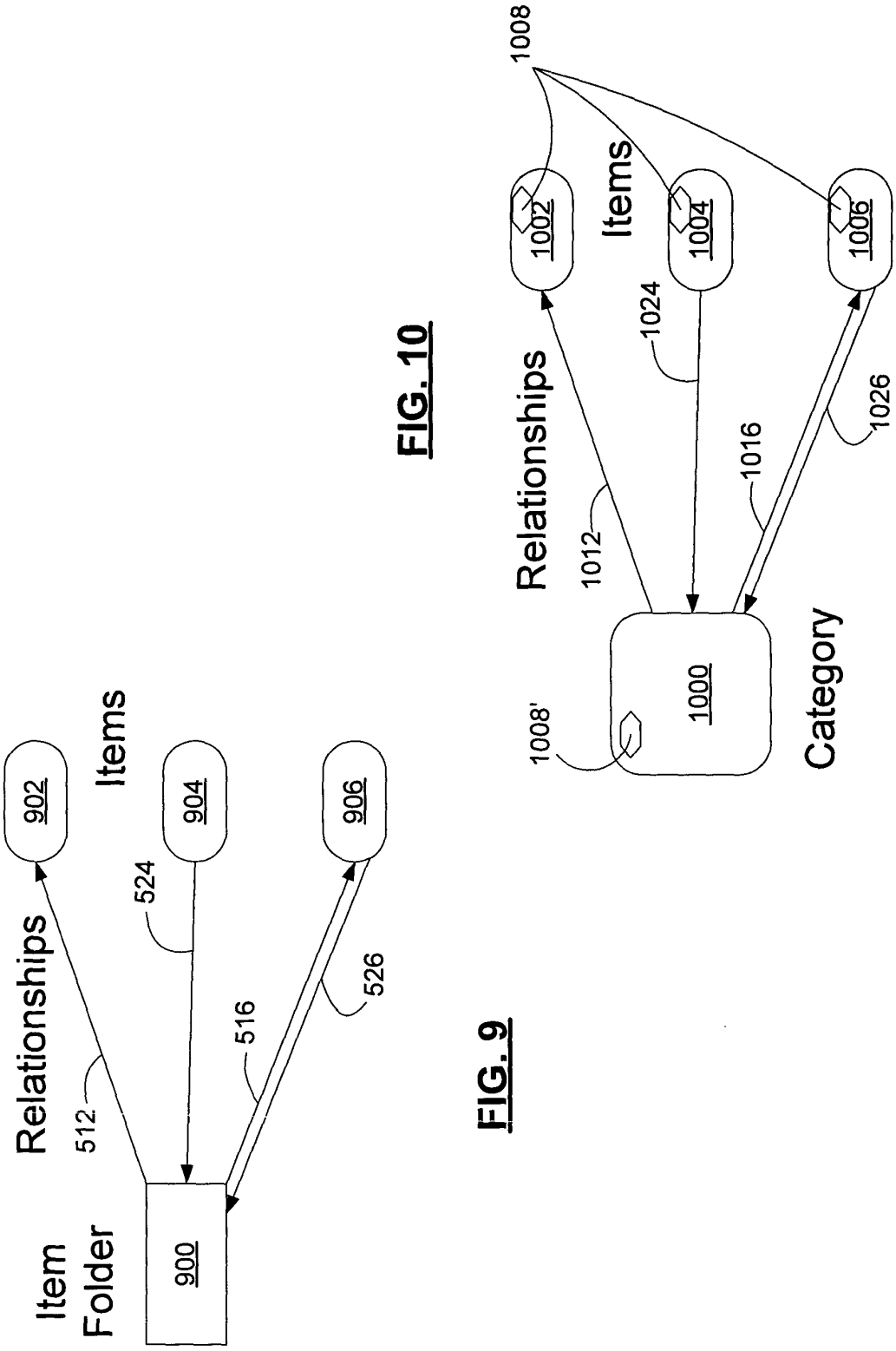

Item Folder
900
Relationships
512
Items
902
904
906
524
516
526
Category
1000
1008'
Relationships
1012
Items
1002
1004
1006
1008
1024
1016
1026

FIG. 9

FIG. 10

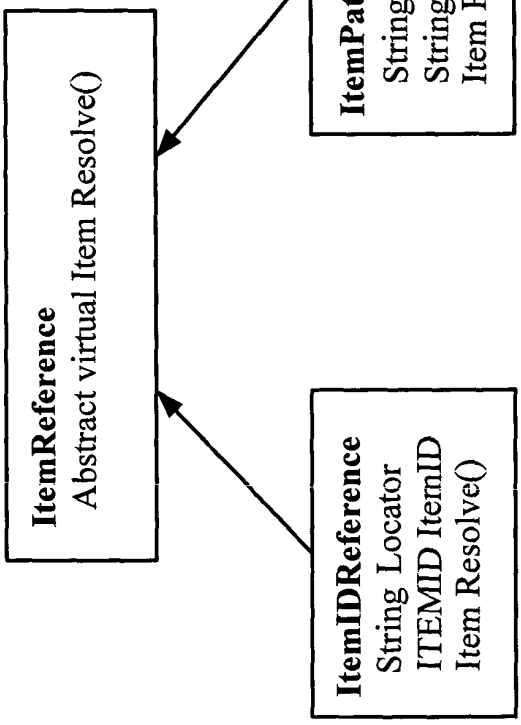

ItemReference
Abstract virtual Item Resolve()
ItemPathReference
String Locator
String Path
Item Resolve()
ItemIDReference
String Locator
ITEMID ItemID
Item Resolve()

FIG. 11

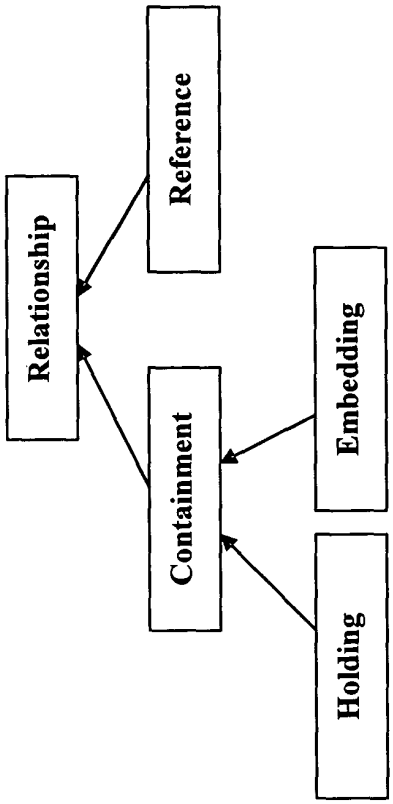

Relationship
Reference
Containment
Embedding
Holding

FIG. 12

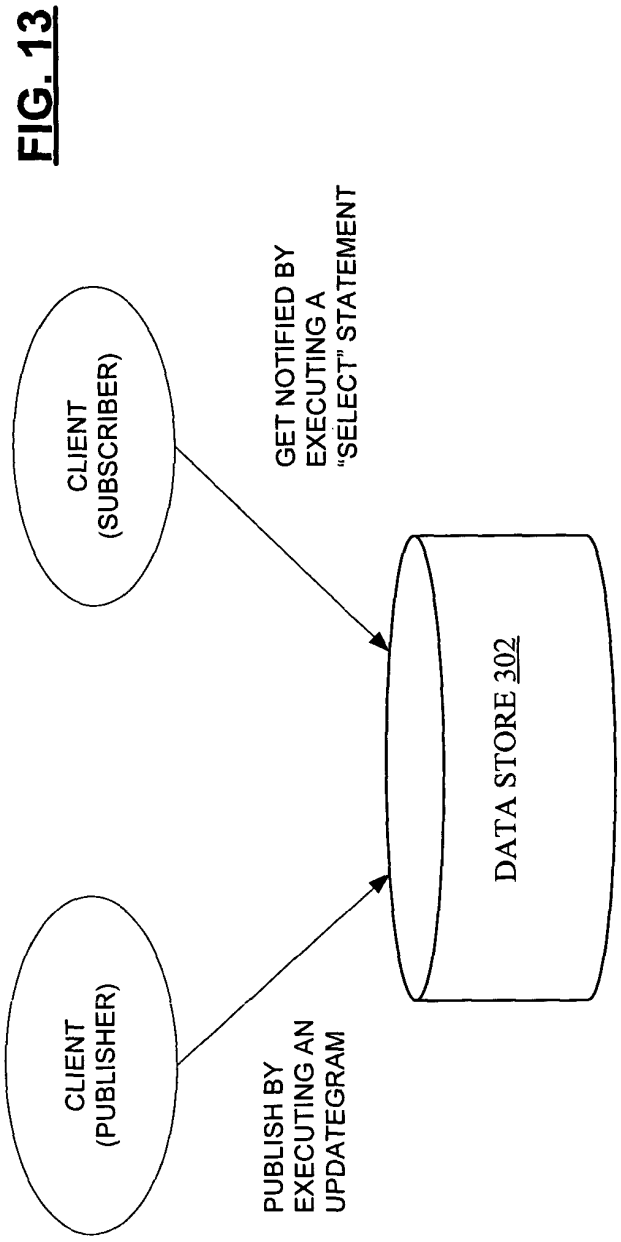

CLIENT (SUBSCRIBER)
GET NOTIFIED BY EXECUTING A "SELECT" STATEMENT
DATA STORE 302
CLIENT (PUBLISHER)
PUBLISH BY EXECUTING AN UPDATEGRAM

FIG. 13

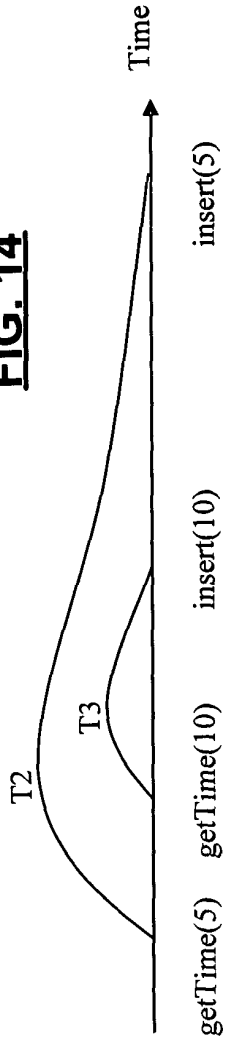

Time
T2
T3
getTime(5)
getTime(10)
insert(10)
insert(5)

FIG. 14

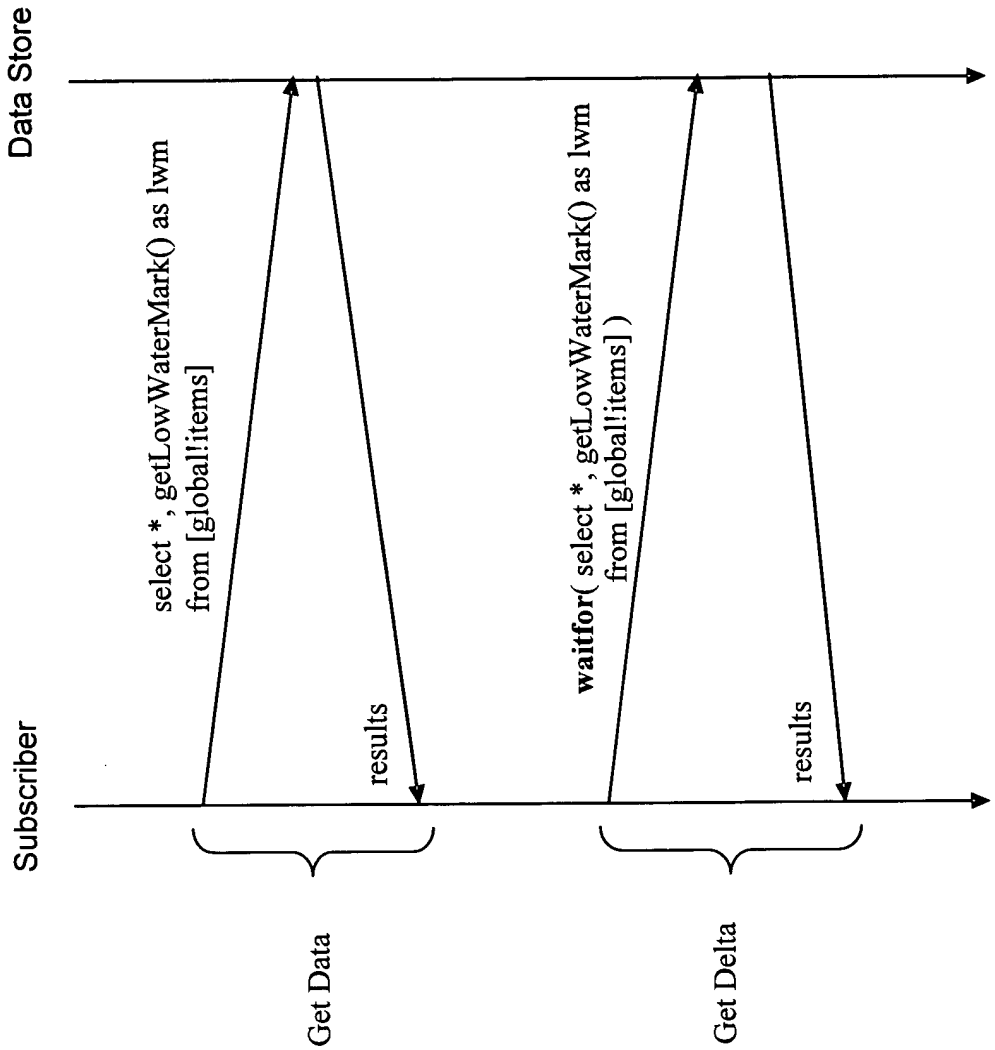

Data Store
Subscriber
select *, getLowWaterMark() as lwm
from [global!items]
results
waitfor(select *, getLowWaterMark() as lwm
from [global!items])
results
Get Data
Get Delta

FIG. 15

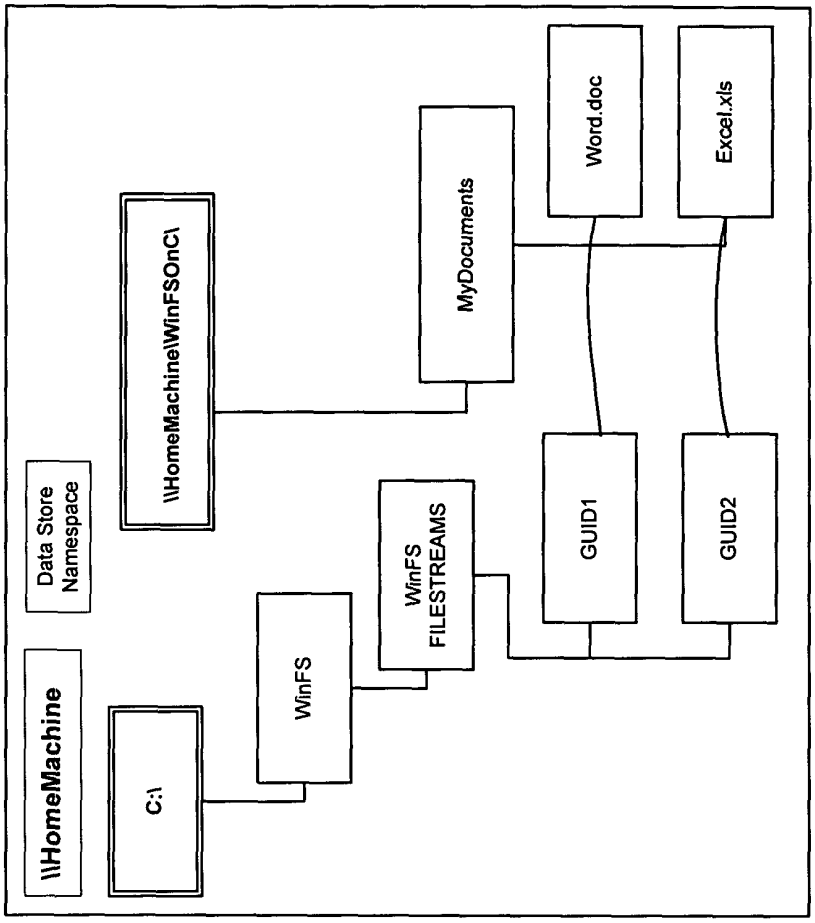

\\HomeMachine
Data Store Namespace
C:\
WinFS
WinFS FILESTREAMS
\\HomeMachine\WinFSOnC\
MyDocuments
GUID1
GUID2
Word.doc
Excel.xls

FIG. 17

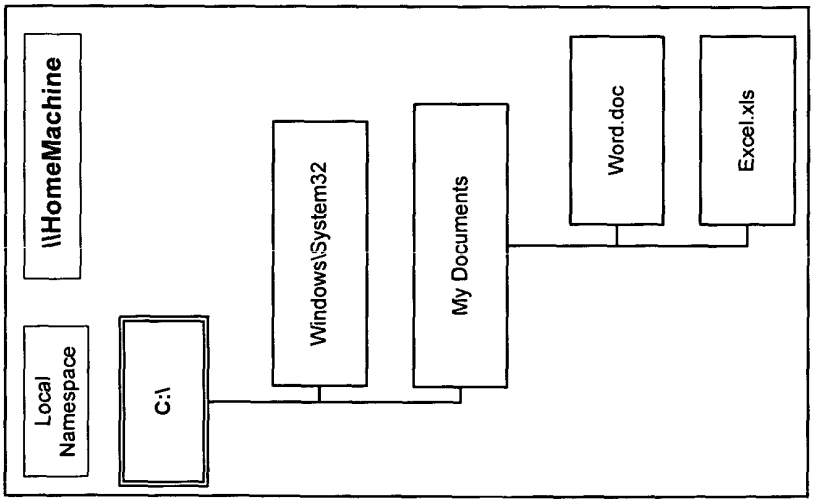

\\HomeMachine
Local Namespace
C:\
Windows\System32
My Documents
Word.doc
Excel.xls

FIG. 16

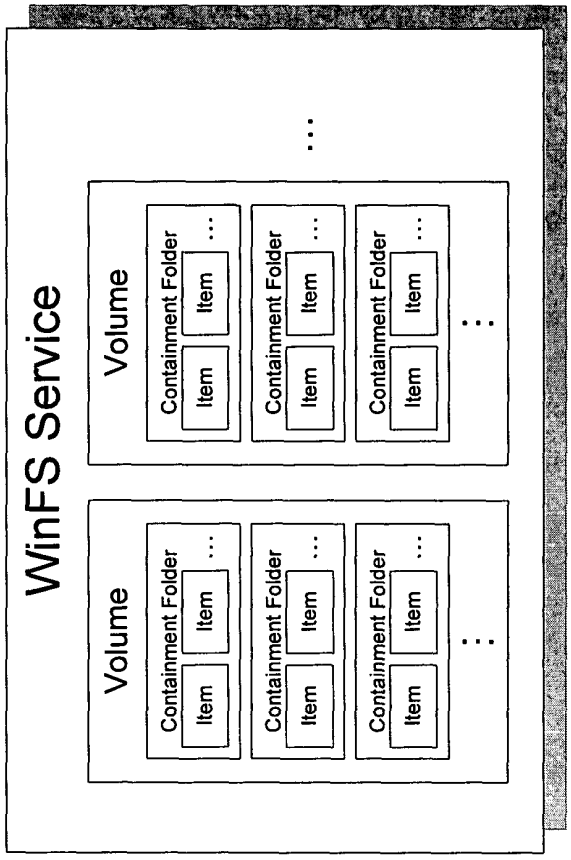

WinFS Service
Volume
Containment Folder
Item
Containment Folder
Item
Containment Folder
Item
Volume
Containment Folder
Item
Containment Folder
Item
Containment Folder
Item

FIG. 18

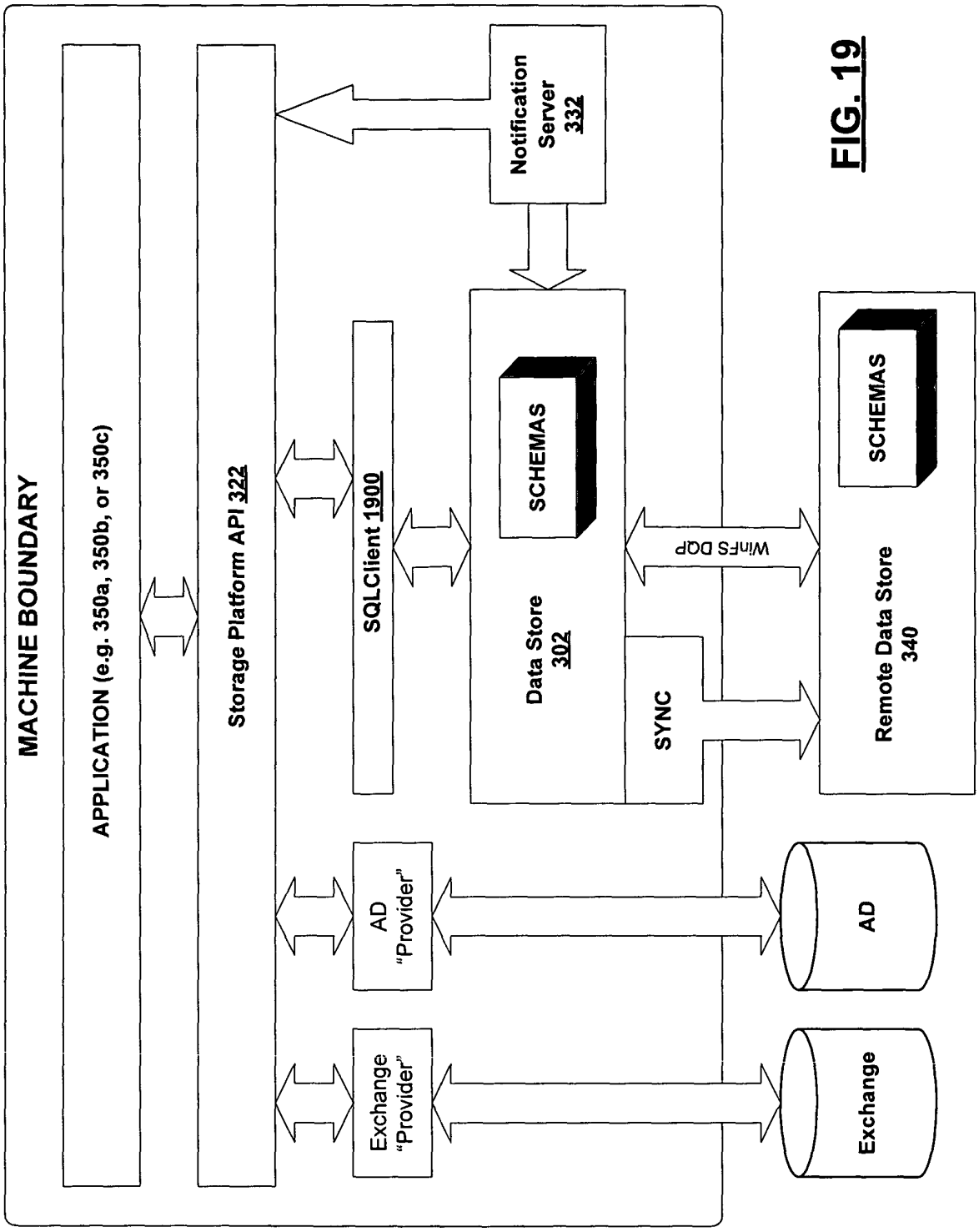

MACHINE BOUNDARY
APPLICATION (e.g. 350a, 350b, or 350c)
Storage Platform API 322
Notification Server 332
SQLClient 1900
SCHEMAS
Data Store 302
WinFS DQP
SYNC
SCHEMAS
Remote Data Store 340
AD "Provider"
AD
Exchange "Provider"
Exchange

FIG. 19

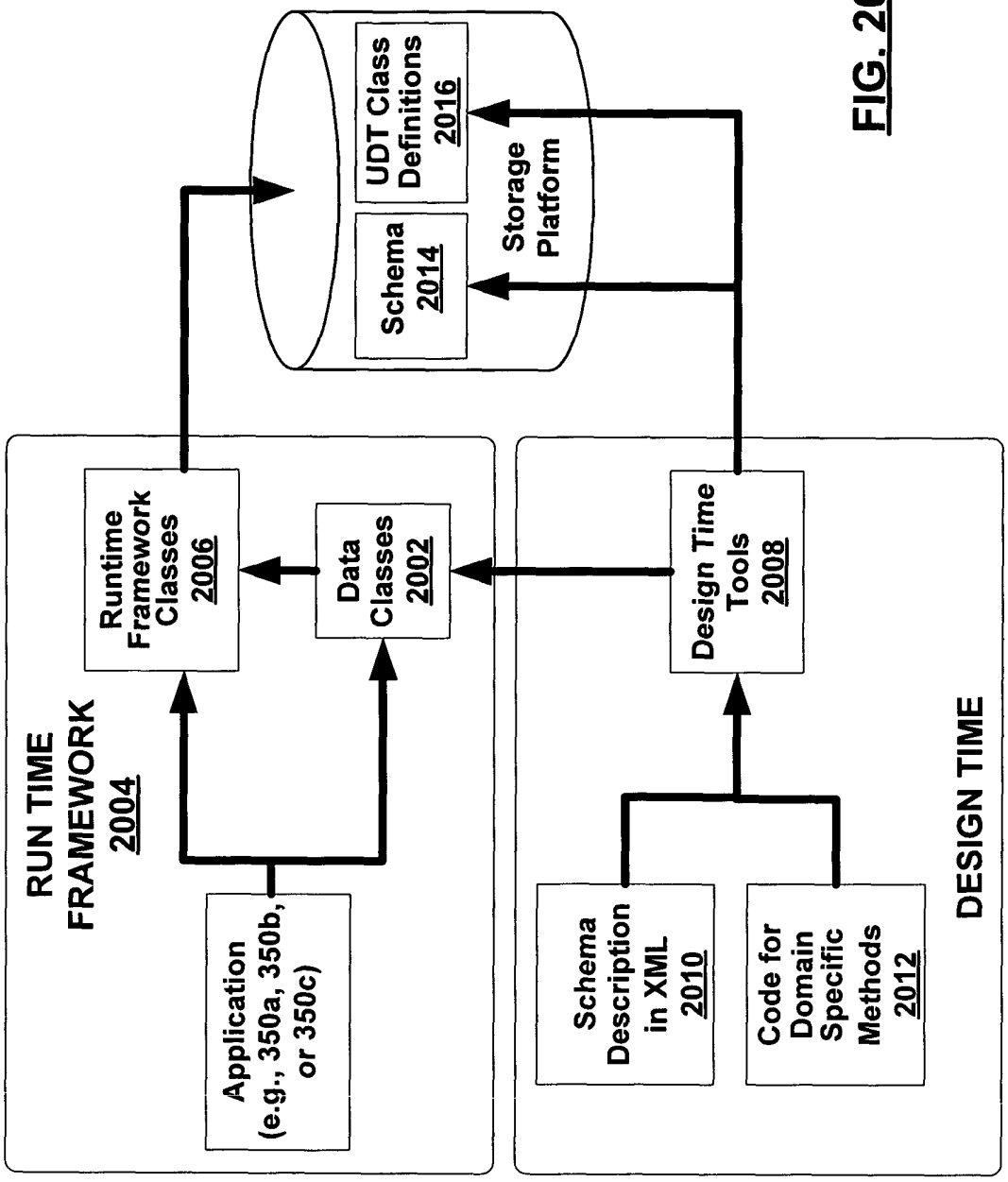

RUN TIME FRAMEWORK 2004
Application (e.g., 350a, 350b, or 350c)
Runtime Framework Classes 2006
Data Classes 2002
DESIGN TIME
Schema Description in XML 2010
Code for Domain Specific Methods 2012
Design Time Tools 2008
Storage Platform
Schema 2014
UDT Class Definitions 2016

FIG. 20

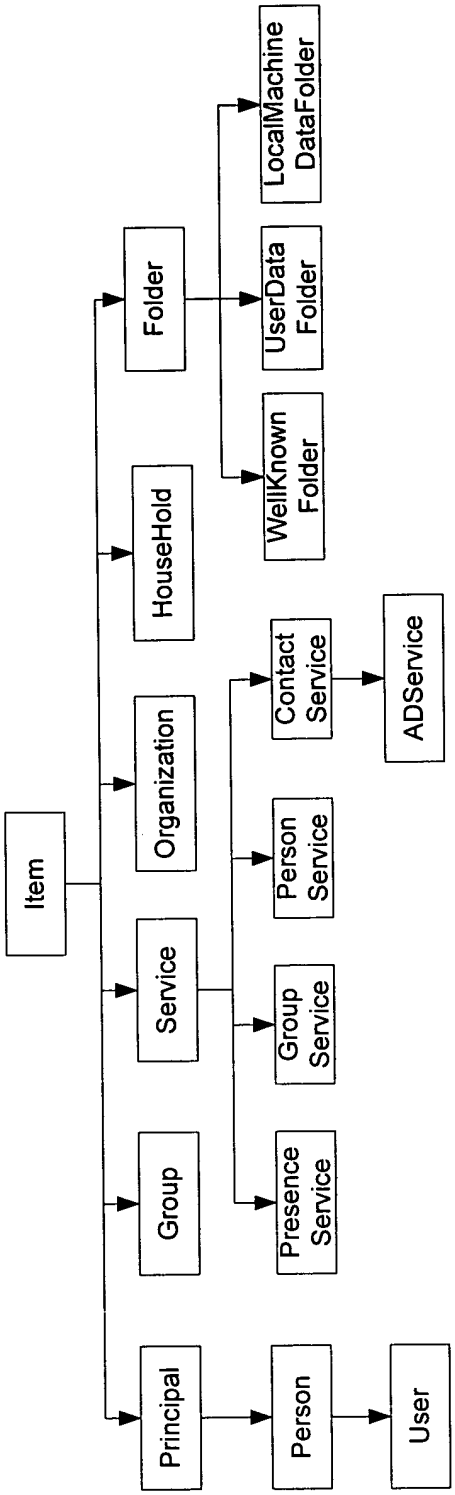

Item
Principal
Person
User
Group
Service
Presence Service
Group Service
Person Service
Organization
Contact Service
ADService
HouseHold
Folder
WellKnown Folder
UserData Folder
LocalMachine DataFolder

FIG. 21A

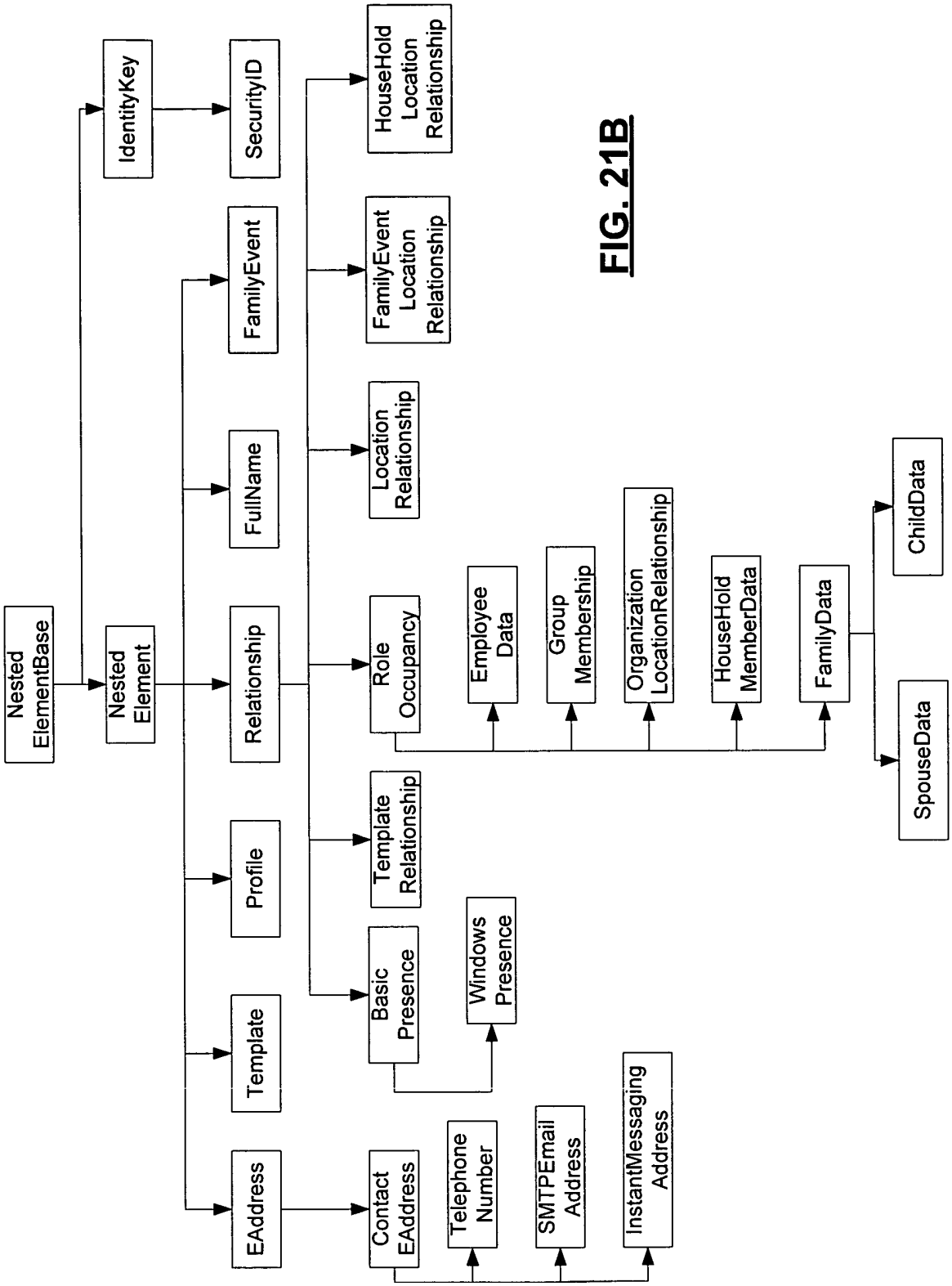

Nested ElementBase
Nested Element
IdentityKey
SecurityID
Relationship
FamilyEvent
FullName
Profile
Template
EAddress
Contact EAddress
Telephone Number
SMTPEmail Address
InstantMessaging Address
Basic Presence
Windows Presence
Template Relationship
Location Relationship
FamilyEvent Location Relationship
HouseHold Location Relationship
Role Occupancy
Employee Data
Group Membership
Organization LocationRelationship
HouseHold MemberData
FamilyData
ChildData
SpouseData

FIG. 21B

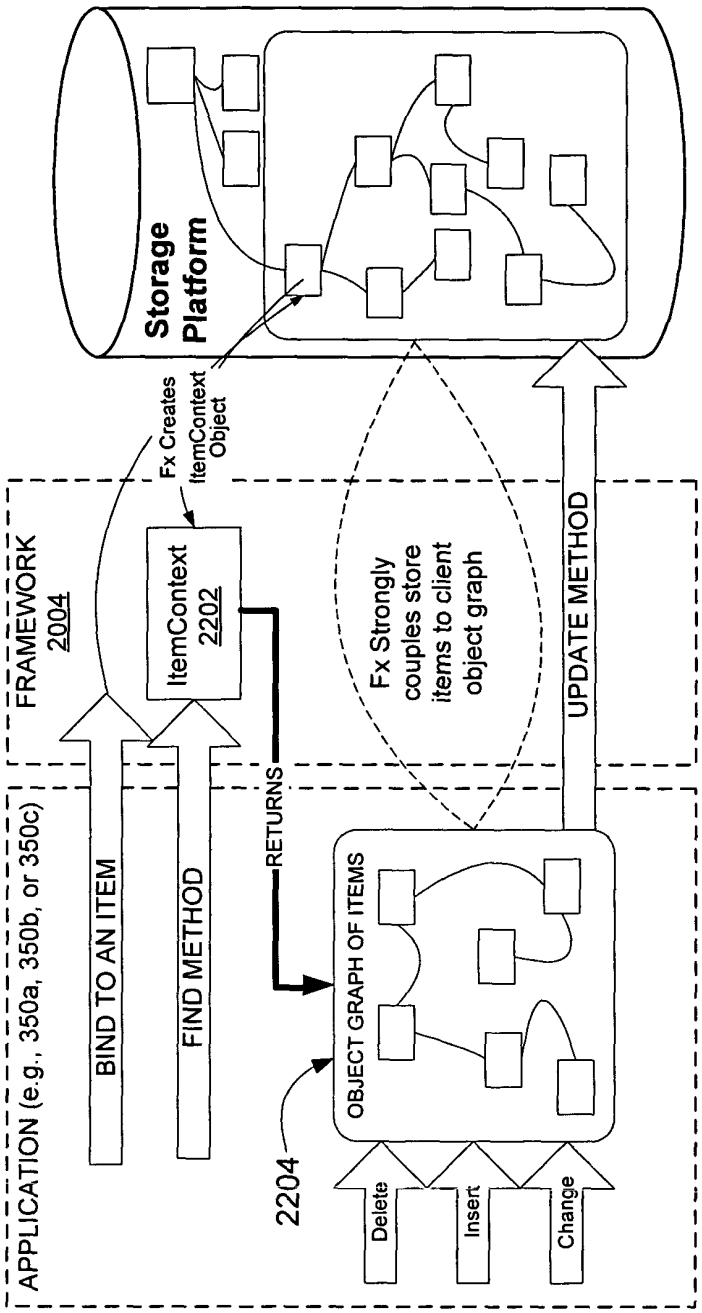

Storage Platform
Fx Creates ItemContext Object
FRAMEWORK 2004
ItemContext 2202
Fx Strongly couples store items to client object graph
UPDATE METHOD
APPLICATION (e.g., 350a, 350b, or 350c)
BIND TO AN ITEM
FIND METHOD
RETURNS
OBJECT GRAPH OF ITEMS
2204
Delete
Insert
Change

FIG. 22

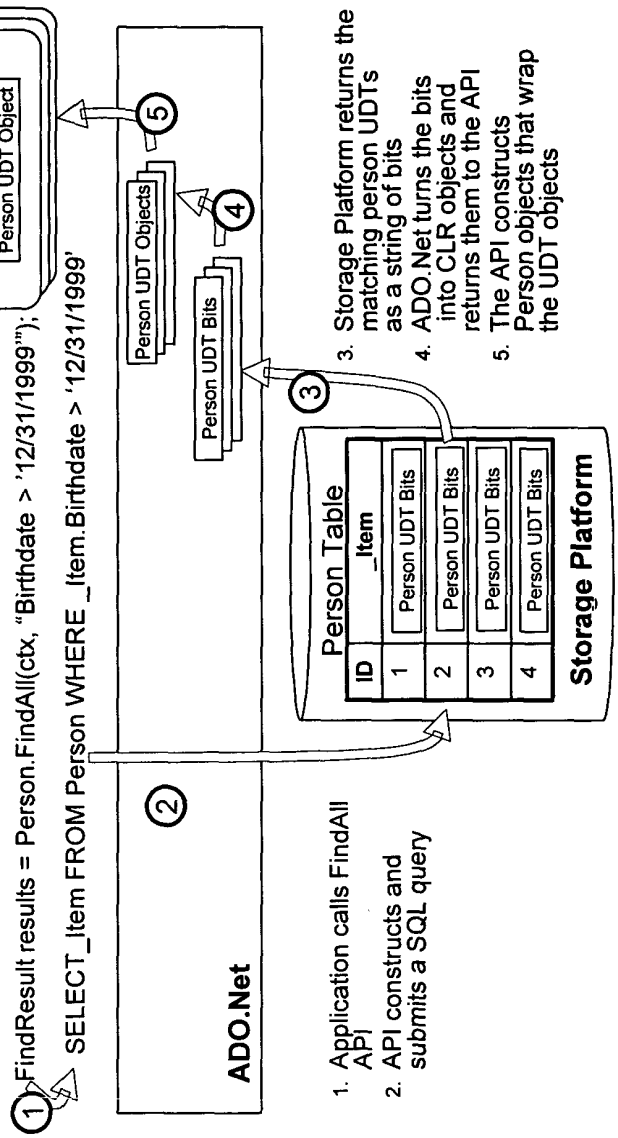

Person Objects
Person UDT Object
Person UDT Objects
Person UDT Bits
ADO.Net
Person Table
ID
_Item
Person UDT Bits
Storage Platform
1. Application calls FindAll API
2. API constructs and submits a SQL query
3. Storage Platform returns the matching person UDTs as a string of bits
4. ADO.Net turns the bits into CLR objects and returns them to the API
5. The API constructs Person objects that wrap the UDT objects
1 FindResult results = Person.FindAll(ctx, "Birthdate > '12/31/1999'");
SELECT _Item FROM Person WHERE _Item.Birthdate > '12/31/1999'

FIG. 23

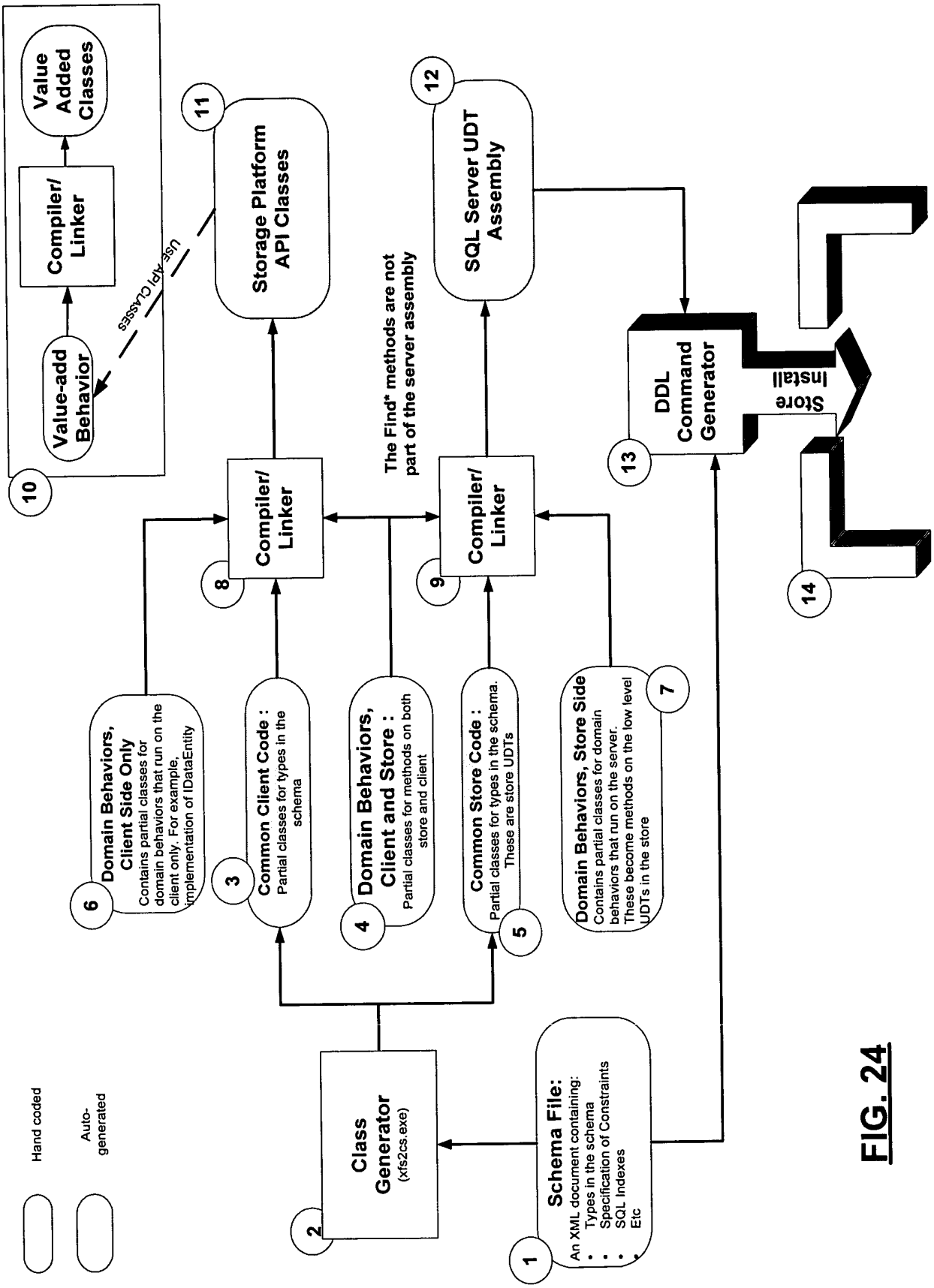
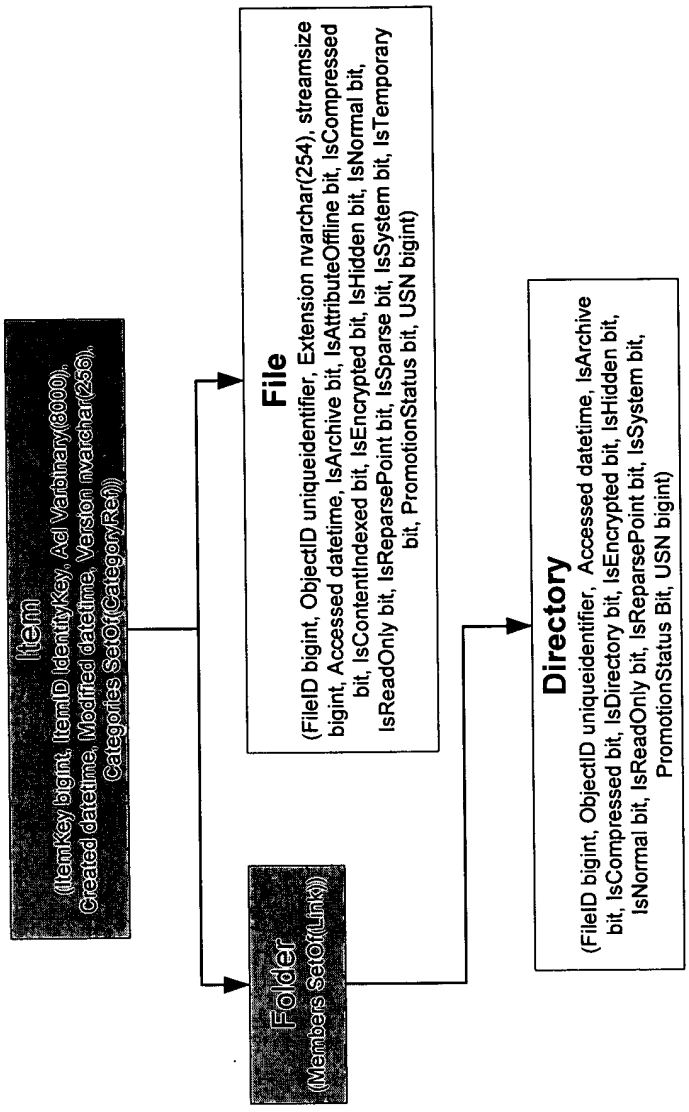

Hand coded
Auto-generated
1
Schema File:
An XML document containing:
Types in the schema
Specification of Constraints
SQL Indexes
Etc
2
Class Generator
(xfs2cs.exe)
3
Common Client Code :
Partial classes for types in the schema
4
Domain Behaviors, Client and Store :
Partial classes for methods on both store and client
5
Common Store Code :
Partial classes for types in the schema.
These are store UDTs
6
Domain Behaviors, Client Side Only
Contains partial classes for domain behaviors that run on the client only. For example, implementation of IDataEntity
7
Domain Behaviors, Store Side
Contains partial classes for domain behaviors that run on the server.
These become methods on the low level UDTs in the store
8
Compiler/ Linker
9
Compiler/ Linker
The Find* methods are not part of the server assembly
10
Value-add Behavior
Use API Classes
Compiler/ Linker
Value Added Classes
11
Storage Platform API Classes
12
SQL Server UDT Assembly
13
DDL Command Generator
14
Store Install

FIG. 24



Note: Grey boxes are types defined in the Base schema, but are shown here for the sake of completeness.

Items Defined in the Files Schema

FIG. 25

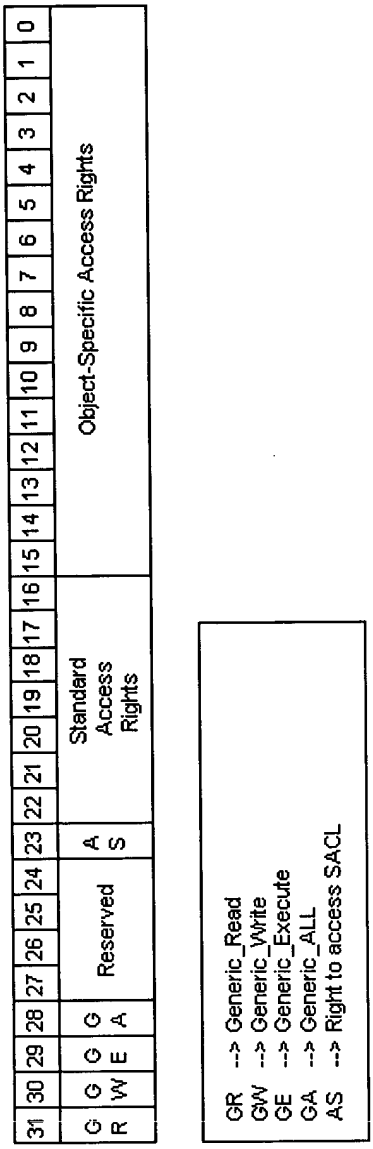

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
GR
GW
GE
GA
Reserved
AS
Standard Access Rights
Object-Specific Access Rights
GR --> Generic_Read
GW --> Generic_Write
GE --> Generic_Execute
GA --> Generic_ALL
AS --> Right to access SACL

FIG. 26

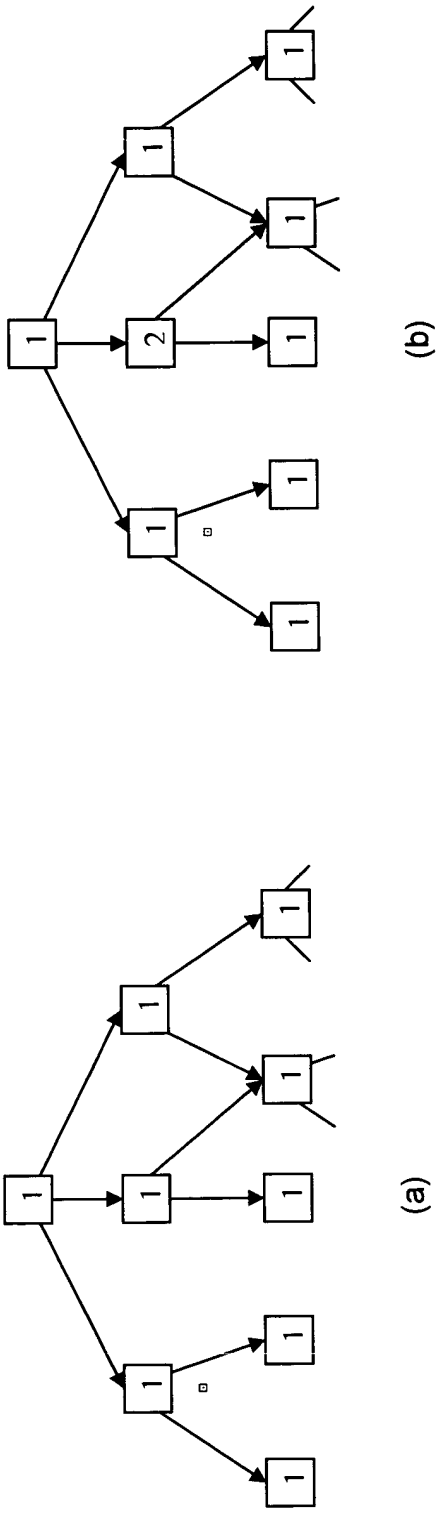


FIG. 27

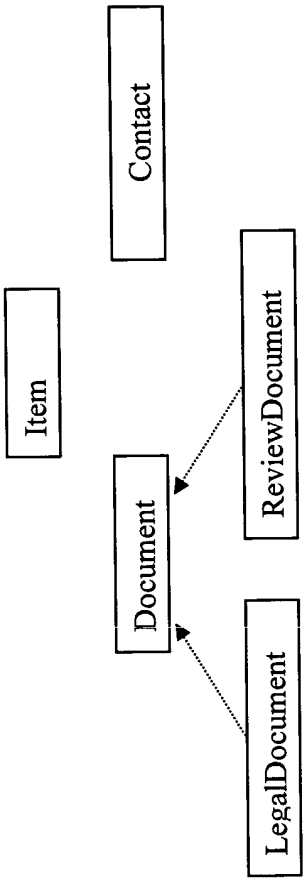

Item
Contact
Document
ReviewDocument
LegalDocument

FIG. 28

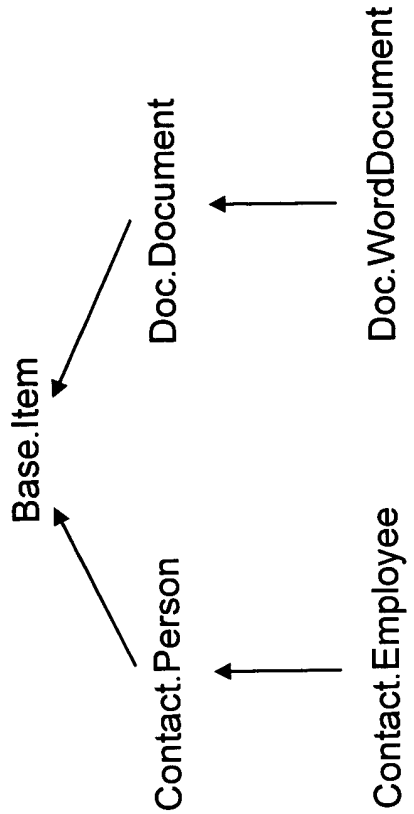

Base.Item
Doc.Document
Doc.WordDocument
Contact.Person
Contact.Employee

FIG. 29